Syllabus-005

Monday, August 15, 2022 2:42 PM

CS/CE 2336.005 - Computer Science II
Mondays & Wednesdays 11:30am-12:45pm ECSS 2.311
Websites: elearning.utdallas.edu (assignment submissions, grades, etc.)
utdallas.box.com/v/cs2336-005-f22

Instructor	Dr. Jey Veerasamy *In-person Office hours:* Mon & Wed 12:45-1:15pm near ECSS 2.311 *Online Office hours (MS Teams utd.link/jey):* Tuesdays 1:30-4pm OR by appointment *Communication:* MS Teams chat (preferred) or email jeyv@utdallas.edu
TA	TBD

Catalog Description

CS/CE 2336 - Computer Science II (3 semester credit hours) Further applications of programming techniques, introducing the fundamental concepts of data structures and algorithms. Topics include recursion, fundamental data structures (including stacks, queues, linked lists, hash tables, trees, and graphs), and algorithmic analysis. Includes comprehensive programming projects. Programming language of choice is Java.

Course Pre-requisite: CS/CE 1337 with a grade of C or better. Prerequisite or Corequisite: CS/CE 2305 with a grade of C or better or (Data Science major and MATH 3315). Credit cannot be received for both CS 2337 and CS/CE 2336.

Student Learning Objectives

Students will be able to implement different data structures using the Java programming language. They will be able use different data structures to program solutions to solve real problems. It will also help them understand algorithmic analysis and complexities. After successful completion of this course, you should be able to:

- Ability to implement recursive algorithms
- Ability to implement linked lists, stacks, and queues
- Ability to implement a binary tree
- Ability to use hash tables and graphs
- Ability to understand algorithmic analysis
- Ability to create a comprehensive programming project
- Ability to implement and use generics/templates

Teaching Philosophy Statement

My goal is to make each class as enjoyable as possible! Since most of my courses are entry-level programming courses, I use project-based learning approach to teach. In other words, I work on a series of real-world scenario-based problems. After discussing each problem in high level, I expect the students to code along with me in the class. If you attend all the classes & be engaged with the content in each session, you will do very well in my courses!

Required Textbooks

Y. Daniel Liang, "Introduction to Java Programming and Data Structures, Comprehensive Version", 11th edition, ISBN 9780134670942

We will be using codio platform to submit & auto-grade the assignments and most activities. Your submissions will be tested against several testcases, similar to ZyLab platform you might have used in CS 1136/1336 courses. You will be promoted to purchase a license ($40 fee) as part of the first assignment/activity we will do in that platform.

You can create your own Java projects in Codio platform. There are several popular IDEs to develop Java programs. You are welcome to use any of them.

- IntelliJ Idea Community Edition https://www.jetbrains.com/idea
- VS Code https://code.visualstudio.com/docs/java/java-tutorial

- Eclipse IDE https://www.eclipse.org/downloads/packages
- Netbeans IDE https://netbeans.apache.org
- Web based IDEs: https://replit.com/new/java, https://www.online-java.com,

Additional Resources

- Intro to Java: http://javabeginnerstutorial.com/core-java
- Java questions: http://stackoverflow.com/questions/tagged/java
- Online Java tutorial: http://introcs.cs.princeton.edu/java/10elements
- Java Docs: https://docs.oracle.com/en/java/index.html
- Another tutorial: https://www.programiz.com/java-programming

Grading Policy

Letter grades will be assigned as follows. I do not plan to the curve the final grades. You have to earn your grade with your work.

97-100: A+	94-97: A	90-94: A-
87-90: B+	84-87: B	80-84: B-
77-80: C+	74-77: C	70-74: C-
67-70: D+	64-67: D	60-64: D-
Below 60: F		

Tentative Course Schedule

Week	Class Activity	Chapters
1	Review of Syllabus Basic Java, Arrays & Strings	3-10
2	Recursion	11
3	Classes, Object-Oriented Design	12
4	Inheritance & Polymorphism	14
5	Abstract Classes & Interfaces	15
6	Linked Lists	16
7	**Test 1 (Testing Center or in class TBD)**	
8	Stacks, Queues	17
9	Exception Handling Generics, Arraylists	19-21
9	Binary Search Trees	22
11	Graphs	24
12	Hashing	26
13	Big-O notation, Algorithmic Efficiency of Data Structures	27
14	Sorting Algorithms	28
15	**Test 2 (Testing Center or in class TBD)**	

Course & Instructor Policies

Course credit is given only for the work assigned in the course schedule. There will be no opportunities for any extra credit in this course. The final grade will be computed as follows.

Tests	30%	There will be 2 tests (Test1: 15% & Test2: 15%) You may need to take each test during the class timings in a computer at the testing center. You

		should reserve your seat as soon as the professor asks you to do so. Each test will contain a few multiple choice questions, fill-in-the-blank questions & one or two coding questions. There will be a few bonus questions to reduce your stress ☺ Any make-up tests will be arranged and scheduled during the same week at the discretion of the instructor. There should be a valid reason (like Dr note, official off-site game participation, etc.) for scheduling make-up tests & you need to coordinate with the instructor in advance. Makeup test due to other scenarios (including missing to book the slot in the testing center) will result in 20% penalty. Best way to prepare for the test will be to attend each class & be engaged, complete the activities and weekly assignments & get your doubts clarified in timely manner.
Assignments	40%	There will be weekly assignments in this course & they will have equal weightage – all of them together will contribute 40% to the final weighted grade. Each weekly assignment will require the students to spend a few hours programming in the computer (I hope to have one comprehensive programming project towards the end, with 2 or more phases - each phase will count as one assignment). Right way to approach any programming assignment is to start right away & ask for help when you get stuck (you can approach the instructor or TA for help). Do not waste several hours trying to fix a small glitch. In simple words, your approach will determine whether the programming assignments provide an enjoyable learning experience or end up as painful & seemingly useless activities. You can also use Discussion Board in elearning to post your issues and get help from others. Please do not share too much code though - everyone should learn on their own with minimal help from others! I will look at the discussion board whenever I can - no promises! I encourage everyone to plan, code & submit the assignments early, since late assignments will NOT be accepted. So, start early! If you cannot complete an assignment due to medical condition, send the Doctor note to the professor using MS Teams chat. You will either get extra time to complete the assignment or you will be exempted from that assignment. If you are worried along the way that you may fail in this course, you are recommended to submit all the missed assignments. While the late submissions may not be graded right away, instructor will take them into account when assigning the final grade.
Activities	30%	There will be several activity items every week to ensure that you are keeping up with the class content (complete tutorials at home, submit class-work or take quiz) - all of them will contribute equally & together they will account for 30% of your final grade. So, you need to attend each lecture to get full points for the activities. Due to the huge # of activities, you will be given a completion grade, that is, all meaningful submissions will get full points. However, we may randomly select a few submissions for complete evaluation & feedback. Multiple attempts will be allowed for quizzes so that you can repeat until you get the full score. If you cannot complete an activity due to medical condition, send the Doctor note to the professor using MS Teams chat. You will be exempted from that activity. Late activity submissions will NOT help you to pass this course, unlike the assignments.

Course & Instructor Policies

You are required to bring a laptop to each class. I expect you to work on one or more coding problems in every class. I expect you to be physically and mentally present in the class. You are NOT allowed to do anything in the laptop or phone unrelated to the class work.

In addition to meeting the instructor before or after the class, you can also visit the instructor or TA during respective virtual office hours. This is preferred approach specifically if you run into project related issues & you need help to progress. Outside the office hours, you can reach out to Computer Science Mentor Center at https://csmc.utdallas.edu. You can also discuss the assignments related issues in the discussion forums in elearning. Please do not share too much code though - enables others to learn on their own with minimal help!

Class Recordings

There is NO plan to record the regular lectures. The instructor may record meetings of this course, under special circumstances. Such recordings will be made available to all students registered for this class if the intent is to supplement the classroom experience. If the instructor or a UTD school/department/office plans any other uses for the recordings, consent of the students identifiable in the recordings is required prior to such use unless an exception is allowed by law.

Comet Creed

This creed was voted on by the UT Dallas student body in 2014. It is a standard that Comets choose to live by and encourage others to do the same: “As a Comet, I pledge honesty, integrity, and service in all that I do.”

Plagiarism has no place in the college education. UTD policies require all the professors to forward all suspicious cases to academic disciplinary committee. So, do not copy the code from others & do not give your code to others.

Academic Support Resources

The information contained in the following link lists the University’s academic support resources for all students. Please see http://go.utdallas.edu/academic-support-resources.

UT Dallas Syllabus Policies and Procedures

The information contained in the following link constitutes the University’s policies and procedures segment of the course syllabus. Please go to go.utdallas.edu/syllabus-policies for these policies.

The descriptions and timelines contained in this syllabus are subject to change at the discretion of the Professor.